

AABD  
2020/2021  
G14  
Meta 3

Diogo Dias Lopes 2018019746  
Leonardo David de Bastos Marques 2019123778  
Ricardo Jorge Loureiro Rodrigues 2017015954

Diogo Dias Lopes 2018019746

Leonardo David de Bastos Marques 2019123778

Ricardo Jorge Loureiro Rodrigues 2017015954

## Procedimentos

create or replace PROCEDURE chk3\_f\_transporta\_frascos(vidTransporte  
number, vidArmazena number, vidlocalVacina varchar , quantidade  
NUMBER,destino varchar, origem varchar,DATAtransporte DATE)

IS

loc number;

vac number;

armaz number;

excecao\_local exception;

Excecao\_vacina exception;

Excecao\_LocalArmazenamento exception;

BEGIN

Select count(local.LOCALVACINACAO) into loc

from local,utente,convocatoria

where local.LOCALVACINACAO=vidlocalVacina;

if loc=0 then

    Raise Excecao\_Local;

end if;

Select count(armazem.id\_armazem) into armaz

from armazem,transporte

where armazem.ID\_ARMAZEM=vidArmazena;

if armaz=0 then

    Raise Excecao\_LocalArmazenamento;

Diogo Dias Lopes 2018019746

Leonardo David de Bastos Marques 2019123778

Ricardo Jorge Loureiro Rodrigues 2017015954

```
INSERT INTO TRANSPORTE
(ID_TRANSPORTE,ID_ARMAZEM,LOCALVACINACAO,N_FRASCOS,DESTI
NO,ORIGEM,DATA)

VALUES
(vidTransporte,vidArmazena,vidlocalVacina,quantidade,destino,origem,DATAtr
ansporte);

EXCEPTION

    WHEN EXCECAO_Local THEN

        RAISE_APPLICATION_ERROR(-20502, 'Local de Vacinação
Inexistente');

    When EXCECAO_LocalArmazenamento THEN

        RAISE_APPLICATION_ERROR(-20508, 'Local de Armazenamento
inexistente');

END;
```

```
create or replace PROCEDURE chk3_h_abre_frasco( vidfrasco number)
IS
    numerodoses number;
    frasc number;
    faberto number;
    tativo number;
    estadofrasco frasco.estado%TYPE;
    excecao_frasco exception;
    excecao_narmazem exception;
    excecao_faberto exception;
    excecao_tativo exception;
```

Diogo Dias Lopes 2018019746

Leonardo David de Bastos Marques 2019123778

Ricardo Jorge Loureiro Rodrigues 2017015954

```
Select count(ID_FRASCO) into frasc  
from frasco  
where ID_FRASCO=vidfrasco  
;
```

```
if frasc=0 then  
Raise excecao_frasco;  
end if;
```

```
Select estado into faberto  
from frasco  
where ID_FRASCO=vidfrasco  
;
```

```
if faberto = 'Aberto' then  
raise excecao_faberto;  
end if;
```

```
select count(id_turno) into tativo  
from turno  
where turno.ATIVO=0;
```

```
if tativo=0 then  
RAISE excecao_tativo;  
end if;
```

Diogo Dias Lopes 2018019746

Leonardo David de Bastos Marques 2019123778

Ricardo Jorge Loureiro Rodrigues 2017015954

```
select estado into estadofrasco
```

```
from frasco
```

```
where id_frasco = vidfrasco;
```

```
IF estadofrasco = 'Aberto' THEN
```

```
    SELECT DISTINCT(seringas.quantidade)into numerodoses
```

```
FROM SERINGAS,FRASCO
```

```
WHERE SERINGAS.ID_FRASCO = vidfrasco;
```

```
UPDATE frasco
```

```
SET n_doses = numerodoses
```

```
WHERE id_frasco = vidfrasco;
```

```
END IF;
```

```
EXCEPTION
```

```
    WHEN EXCECAO_frasco THEN
```

```
        RAISE_APPLICATION_ERROR(-20511, 'Frasco Inexistente');
```

```
    When EXCECAO_narmazem THEN
```

```
        RAISE_APPLICATION_ERROR(-20512, 'Frasco não está no local de  
armazenamento.');
```

```
    When EXCECAO_faberto THEN
```

```
        RAISE_APPLICATION_ERROR(-20513, 'Frasco já foi aberto.');
```

```
    When EXCECAO_tativo THEN
```

```
        RAISE_APPLICATION_ERROR(-20513, 'Não existe um truno de  
vacinação ativo.');
```

```
END;
```

Diogo Dias Lopes 2018019746

Leonardo David de Bastos Marques 2019123778

Ricardo Jorge Loureiro Rodrigues 2017015954

create or replace PROCEDURE

CHK3\_PROC\_A2018019746 (idenf number,numpess number)

IS

datainic date;

datafim date;

tempoenf number;

tempoturn number;

tativo number;

estado number;

enferm number;

Excecao\_enfermeiro Exception;

Excecao\_trab Exception;

BEGIN

Select count(id\_enfermeiro) into enferm

from ENFERMEIRO

where ID\_ENFERMEIRO=idenf;

if enferm=0 then

Raise Excecao\_enfermeiro;

end if;

select ENFERMEIRO.ESTADO\_ENF into estado

from enfermeiro

where ID\_ENFERMEIRO=idenf

;

if estado != 'Trabalhar' then

raise excecao\_trab;

Diogo Dias Lopes 2018019746

Leonardo David de Bastos Marques 2019123778

Ricardo Jorge Loureiro Rodrigues 2017015954

```
select TURNO.DATAINICIOTURNO into datainic  
from turno  
where turno.ID_ENFERMEIRO=idenf  
and ativo=1;
```

```
select turno.DATAFIMTURNO into datafim  
from turno  
where turno.ID_ENFERMEIRO=idenf  
and ativo=1;
```

```
tempoturn:=to_number(datafim-datainic);  
tempoenf:=CHK3_FUNC_A2018019746(idenf,numpass);
```

```
select ativo into tativo  
from turno  
where turno.ID_ENFERMEIRO=idenf  
and ativo=1;
```

```
if (tempoenf >= tempoturn and tativo=1)then  
UPDATE turno  
SET ativo = 0  
where turno.ID_ENFERMEIRO=idenf;  
end if;
```

EXCEPTION

Diogo Dias Lopes 2018019746  
Leonardo David de Bastos Marques 2019123778  
Ricardo Jorge Loureiro Rodrigues 2017015954



```
    RAISE_APPLICATION_ERROR(-20504, 'Enfermeiro Inexistente');  
    When EXCECAO_trab THEN  
        RAISE_APPLICATION_ERROR(-20520, 'Enfermeiro não está a  
trabalhar');
```

```
END;
```

```
create or replace PROCEDURE CHK3_PROC_A2019123778 ( vidutente  
number)
```

```
IS
```

```
ut number;
```

```
numerodoses number;
```

```
EXCECAO_utente exception;
```

```
BEGIN
```

```
Select count(utente.id_utente) into ut  
from utente  
where utente.ID_UTENTE=vIdUtente;
```

```
if ut=0 then
```

```
    Raise Excecao_utente;
```

```
end if;
```

```
select distinct(COUNT(toma.ID_TOMA)) into numerodoses  
from TOMA,UTENTE  
where toma.id_utente = vidutente;
```

Diogo Dias Lopes 2018019746

Leonardo David de Bastos Marques 2019123778

Ricardo Jorge Loureiro Rodrigues 2017015954

UPDATE utente

SET NDOSESTOMADAS = numerodoses

where utente.id\_utente = vidutente;

EXCEPTION

When EXCECAO\_utente THEN

RAISE\_APPLICATION\_ERROR(-20501, 'Utente Inexistente');

END;

create or replace PROCEDURE ck3\_j\_cria\_turno\_vacinacao(vidturno number  
,vidlocal varchar, videnfermeiro NUMBER, datainicio DATE, datafim DATE,  
tempo\_por\_vacina NUMBER)

IS

numerovacinasdisponiveis number;

numeroutentes number;

estadoenfermeitov enfermeiro.ESTADO\_ENF%type;

BEGIN

select distinct(enfermeiro.estado\_enf) into estadoenfermeitov

from enfermeiro

where enfermeiro.id\_enfermeiro = videnfermeiro;

select distinct(local.NUMVACINASDISPONIVEIS) into  
numerovacinasdisponiveis

Diogo Dias Lopes 2018019746

Leonardo David de Bastos Marques 2019123778

Ricardo Jorge Loureiro Rodrigues 2017015954

where turno.localvacinacao= local.localvacinacao and turno.localvacinacao =  
vidlocal;

select distinct(count(utente.id\_utente)) into numeroutentes

from local,concelho,utente,turno

where local.id\_concelho= concelho.id\_concelho and concelho.id\_concelho =  
utente.id\_concelho and turno.localvacinacao= local.localvacinacao and  
turno.localvacinacao = vidlocal;

if (numerovacinasdisponiveis >numeroutentes and estadoenfermeitov = 'Nao  
trabalhar' ) then

INSERT INTO turno

(ID\_TURNO,ID\_enfermeiro,LOCALVACINACAO,DATAINICIOTURNO,DATA  
FIMTURNO,ATIVO)

VALUES (vidturno,videnfermeiro,vidlocal,datainicio,datafim,'1');

end if;

END;

### Funções

create or replace FUNCTION CHK3\_FUNC\_A2018019746(codenf  
number,numpac number)

RETURN number

IS

estado enfermeiro.estado\_enf%type;

tempovacinao number;

ritmoenf number;

enferm number;

Excecao\_enfermeiro Exception;

Diogo Dias Lopes 2018019746

Leonardo David de Bastos Marques 2019123778

Ricardo Jorge Loureiro Rodrigues 2017015954

BEGIN

```
Select count(id_enfermeiro) into enferm  
from ENFERMEIRO  
where ID_ENFERMEIRO=codenf;
```

```
if enferm=0 then  
    Raise Excecao_enfermeiro;  
end if;
```

```
select ENFERMEIRO.ESTADO_ENF into estado  
from enfermeiro  
where ID_ENFERMEIRO=codenf  
;
```

```
if estado != 'Trabalhar' then  
    raise excecao_trab;  
end if;
```

```
SELECT enfermeiro.RITMO_VACINACAO INTO ritmoenf  
FROM ENFERMEIRO  
where ID_ENFERMEIRO=codenf;
```

```
tempovacinao:=ritmoenf*numpac;  
RETURN tempovacinao;
```

EXCEPTION

Diogo Dias Lopes 2018019746

Leonardo David de Bastos Marques 2019123778

Ricardo Jorge Loureiro Rodrigues 2017015954

```
    RAISE_APPLICATION_ERROR(-20504, 'Enfermeiro Inexistente');  
  
    When EXCECAO_trab THEN  
  
        RAISE_APPLICATION_ERROR(-20520, 'Enfermeiro não está a  
trabalhar');  
  
END;  
  
create or replace FUNCTION CHK3_FUNC_A2019123778 (vIdFrasco number)  
RETURN frasco.data_aberto%TYPE  
IS  
    frasc number;  
    Datafrasco frasco.data_aberto%TYPE;  
    estadofrasco frasco.estado%TYPE;  
    excecao_frasco exception;  
BEGIN  
  
    Select count(id_frasco) into frasc  
    from frasco  
    where ID_frasco=vidfrasco;  
  
    if frasc=0 then  
        Raise Excecao_frasco;  
    end if;  
  
    select estado into estadofrasco  
    from frasco  
    where id_frasco = vidfrasco;  
  
    IF estadofrasco = 'Aberto' THEN
```

Diogo Dias Lopes 2018019746

Leonardo David de Bastos Marques 2019123778

Ricardo Jorge Loureiro Rodrigues 2017015954

FROM frasco

where id\_frasco = vidfrasco;

END IF;

RETURN Datafrasco;

EXCEPTION

When EXCECAO\_frasco THEN

RAISE\_APPLICATION\_ERROR(-20521, 'Frasco Inexistente');

END;

create or replace FUNCTION ck3\_a\_fase\_vacinacao(codutente VARCHAR)

RETURN fase.datainicioturno%TYPE

IS

datafunction fase.datainicioturno%TYPE;

n number;

vac number;

Excecao\_utente Exception;

EXCECAO\_VACINATOMADA Exception;

BEGIN

Select count(utente.id\_utente) into n

from utente

where ID\_UTENTE=codutente;

Diogo Dias Lopes 2018019746

Leonardo David de Bastos Marques 2019123778

Ricardo Jorge Loureiro Rodrigues 2017015954

```
if n=0 then
    Raise Excecao_utente;
end if;

Select count(toma.id_toma) into vac
from toma
where toma.ID_UTENTE=codutente;

if vac>0 then
    Raise Excecao_vacinatomada;
end if;

SELECT DISTINCT(fase.datainicioturno) into datafunction
FROM FASE, UTENTE, PROFISSAO
WHERE fase.id_profissao = profissao.id_profissao AND profissao.id_profissao
= utente.id_profissao AND utente.ID_UTENTE = codutente;

RETURN datafunction;

EXCEPTION
    WHEN EXCECAO_utente THEN                                --LANÇA
EXCECOES
    RAISE_APPLICATION_ERROR(-20501, 'Utente Inexistente');
When EXCECAO_VACINATOMADA THEN
    RAISE_APPLICATION_ERROR(-20505, 'Vacina já tomada');
END;
```

RETURN seringas.quantidade%TYPE

IS

vacinasdadas seringas.quantidade%TYPE;

loc number;

vac number;

Excecao\_Local Exception;

EXCECAO\_VACINA Exception;

BEGIN

Select count(LOCALVACINACAO) into loc

from local

where LOCALVACINACAO=nomeLocal;

if loc=0 then

    Raise Excecao\_Local;

end if;

Select count(vacina.id\_vacina) into vac

from vacina

where vacina.ID\_vacina=vIdVacina;

if vac=0 then

    Raise Excecao\_vacina;

end if;

SELECT SUM(SERINGAS.QUANTIDADE) into vacinasdadas

FROM TOMA,UTENTE,CONCELHO,LOCAL,SERINGAS

Diogo Dias Lopes 2018019746

Leonardo David de Bastos Marques 2019123778

Ricardo Jorge Loureiro Rodrigues 2017015954



```
WHERE local.id_concelho = concelho.id_concelho and  
utente.id_concelho= concelho.id_concelho and toma.id_utente =  
utente.id_utente AND TOMA.ID_TOMA = SERINGAS.ID_TOMA AND  
LOCALVACINACAO = nomeLocal;
```

```
RETURN vacinasdadas;
```

```
EXCEPTION
```

```
    WHEN EXCECAO_Local THEN                                --LANÇA  
EXCECOES
```

```
    RAISE_APPLICATION_ERROR(-20502, 'Local Inexistente');
```

```
When EXCECAO_VACINA THEN
```

```
    RAISE_APPLICATION_ERROR(-20503, 'Vacina Inexistente');
```

```
END;
```

```
create or replace FUNCTION ck3_d_vacinas_disponiveis (nomeLocal varchar,  
vIdVacina number)
```

```
RETURN number
```

```
IS
```

```
vacinasdisponiveis number;
```

```
    loc number;
```

```
vac number;
```

```
Excecao_Local Exception;
```

```
EXCECAO_VACINA Exception;
```

```
BEGIN
```

```
Select count(LOCALVACINACAO) into loc
```

```
from local
```

```
where LOCALVACINACAO=nomeLocal;
```

Diogo Dias Lopes 2018019746

Leonardo David de Bastos Marques 2019123778

Ricardo Jorge Loureiro Rodrigues 2017015954

```
        Raise Excecao_Local;  
    end if;
```

```
    Select count(vacina.id_vacina) into vac  
    from vacina  
    where vacina.ID_vacina=vIdVacina;
```

```
    if vac=0 then  
        Raise Excecao_vacina;  
    end if;
```

```
    SELECT local.numvacinasdisponiveis into vacinasdisponiveis  
    FROM LOCAL  
    WHERE LOCALVACINACAO = nomeLocal;
```

```
    RETURN vacinasdisponiveis;
```

```
EXCEPTION
```

```
    WHEN EXCECAO_Local THEN                                --LANÇA  
EXCECOES
```

```
        RAISE_APPLICATION_ERROR(-20502, 'Local Inexistente');
```

```
    When EXCECAO_VACINA THEN
```

```
        RAISE_APPLICATION_ERROR(-20503, 'Vacina Inexistente');
```

```
END;
```

```
create or replace FUNCTION ck3_e_data_ultima_convocatoria(vIdUtente  
number)
```

```
RETURN fase.datainicioturno%TYPE
```

Diogo Dias Lopes 2018019746

Leonardo David de Bastos Marques 2019123778

Ricardo Jorge Loureiro Rodrigues 2017015954

```
DATAX convocatoria.data_convocatoria%TYPE;
```

```
ut number;
```

```
loc number;
```

```
vac number;
```

```
armaz number;
```

```
Excecao_utente Exception;
```

```
Excecao_Local Exception;
```

```
EXCECAO_VACINA Exception;
```

```
Excecao_LocalArmazenamento EXCEPTION;
```

```
BEGIN
```

```
    Select count(utente.id_utente) into ut
```

```
    from utente
```

```
    where utente.ID_UTENTE=vIdUtente;
```

```
    if ut=0 then
```

```
        Raise Excecao_utente;
```

```
    end if;
```

```
    Select count(local.LOCALVACINACAO) into loc
```

```
    from local,utente,convocatoria
```

```
    where utente.ID_UTENTE=vIdUtente
```

```
    and utente.ID_UTENTE=convocatoria.ID_UTENTE
```

```
    and convocatoria.LOCALVACINACAO=local.LOCALVACINACAO;
```

```
    if loc=0 then
```

Diogo Dias Lopes 2018019746

Leonardo David de Bastos Marques 2019123778

Ricardo Jorge Loureiro Rodrigues 2017015954

end if;

```
Select count(vacina.id_vacina) into vac
from utente,toma,seringas,frasco,lotas,vacina
where utente.Id_Utente=vIdUtente
and utente.ID_UTENTE=toma.ID_UTENTE
and toma.ID_TOMA=seringas.ID_TOMA
and seringas.ID_FRASCO=frasco.ID_FRASCO
and frasco.ID_LOTE=lotas.ID_LOTE
and lotas.ID_VACINA=vacina.ID_VACINA
;
```

```
if vac=0 then
    Raise Excecao_vacina;
end if;
```

```
Select count(armazem.id_armazem) into armaz
from utente,toma,seringas,frasco,lotas,armazem
where utente.Id_Utente=vIdUtente
and utente.ID_UTENTE=toma.ID_UTENTE
and toma.ID_TOMA=seringas.ID_TOMA
and seringas.ID_FRASCO=frasco.ID_FRASCO
and frasco.ID_LOTE=lotas.ID_LOTE
and lotas.ID_ARMAZEM=armazem.ID_ARMAZEM;
```

```
if armaz=0 then
    Raise Excecao_LocalArmazenamento;
```

Diogo Dias Lopes 2018019746

Leonardo David de Bastos Marques 2019123778

Ricardo Jorge Loureiro Rodrigues 2017015954

```
SELECT data_convocatoria INTO DATA_X
FROM UTENTE, CONVOCATORIA
WHERE UTENTE.ID_UTENTE = CONVOCATORIA.ID_UTENTE AND
CONVOCATORIA.ID_UTENTE = vIdUtente ;

RETURN DATA_X;

EXCEPTION
When EXCECAO_utente THEN
    RAISE_APPLICATION_ERROR(-20501, 'Utente Inexistente');
When EXCECAO_Local THEN
    RAISE_APPLICATION_ERROR(-20502, 'Local Inexistente');
When EXCECAO_VACINA THEN
    RAISE_APPLICATION_ERROR(-20503, 'Vacina Inexistente');
When EXCECAO_LocalArmazenamento THEN
    RAISE_APPLICATION_ERROR(-20508, 'Local de Armazenamento
inexistente');
END;

create or replace FUNCTION ck3_i_data_proxima_dose (vIdUtente number)
RETURN fase.data_inicio_turno%TYPE
IS
DATA_X toma.data_toma%TYPE;
DATA_FINAL toma.data_toma%TYPE;
tempo number;
n number;
```

Diogo Dias Lopes 2018019746

Leonardo David de Bastos Marques 2019123778

Ricardo Jorge Loureiro Rodrigues 2017015954

excecao\_utente exception;

excecao\_ndose exception;

BEGIN

Select count(utente.id\_utente) into n

from utente

where ID\_UTENTE=vIdUtente;

if n=0 then

    Raise excecao\_utente;

end if;

Select utente.ndosestomadas into dosestomadas

from utente

where ID\_UTENTE=vIdUtente;

if dosestomadas=0 then

    Raise excecao\_ndose;

end if;

SELECT DATA\_TOMA INTO DATA\_X

FROM UTENTE,TOMA

WHERE UTENTE.ID\_UTENTE = TOMA.ID\_UTENTE AND  
TOMA.ID\_UTENTE = vIdUtente ;

SELECT TEMPO\_INTERVALO INTO tempo

FROM VACINA,TOMA,REACAO

Diogo Dias Lopes 2018019746

Leonardo David de Bastos Marques 2019123778

Ricardo Jorge Loureiro Rodrigues 2017015954

```
WHERE VACINA.ID_VACINA = REACAO.id_vacina  
AND TOMA.ID_TOMA = REACAO.ID_REACAO and TOMA.ID_UTENTE =  
vIdUtente;
```

```
datafinal:=datax+tempo;
```

```
RETURN DATAFINAL;
```

```
EXCEPTION
```

```
When EXCECAO_utente THEN
```

```
    RAISE_APPLICATION_ERROR(-20501, 'Utente Inexistente');
```

```
WHEN EXCECAO_ndose THEN
```

```
    RAISE_APPLICATION_ERROR(-20515, 'Utente ainda não tomou  
nenhuma dose');
```

```
END;
```

## Triggers

```
create or replace TRIGGER chk3_k_tg_apos_toma
```

```
AFTER INSERT ON toma
```

```
FOR EACH ROW
```

```
DECLARE
```

```
    rea number;
```

```
    LOCALA LOCAL.LOCALVACINACAO%TYPE;
```

```
    SERINGAA SERINGAS.ID_SERINGA%TYPE;
```

```
BEGIN
```

Diogo Dias Lopes 2018019746

Leonardo David de Bastos Marques 2019123778

Ricardo Jorge Loureiro Rodrigues 2017015954

from reacao,toma

where reacao.ID\_TOMA=:New.id\_toma;

if rea > 0 then

UPDATE frasco --ATUALIZA O ESTADO Dos frascos

SET ESTADO ='inutilzavel';

end if;

END;

create or replace TRIGGER chk3\_1\_tg\_convoca\_proximo

after UPDATE OF REJEITOU ON UTENTE

FOR EACH ROW

DECLARE

IDNCONV UTENTE.ID\_UTENTE%TYPE;

convoc CONVOCATORIA.ID\_CONVOCATORIA%type;

BEGIN

SELECT UTENTE.ID\_UTENTE into IDNCONV

FROM UTENTE

WHERE CONVOCADO=0

and rownum = 1

;

select CONVOCATORIA.ID\_CONVOCATORIA into convoc

from convocatoria

where convocatoria.id\_utente=IDNCONV;

Diogo Dias Lopes 2018019746

Leonardo David de Bastos Marques 2019123778

Ricardo Jorge Loureiro Rodrigues 2017015954



```
UPDATE CONVOCATORIA
SET id_utente = IDNCONV
where id_convocatoria=convoc;
end if;
END;

create or replace TRIGGER CHK3_Q_TRIG_A2018019746
after update of estado_enf ON enfermeiro
FOR EACH ROW
DECLARE

BEGIN

if :new.estado_enf = 'não trabalhar' then
UPDATE turno                                --ATUALIZA O ESTADO DA SERINGA
UTILIZADA
SET ativo = 0
WHERE turno.id_enfermeiro = :new.id_enfermeiro;
end if;

END;

create or replace TRIGGER CHK3_Q_TRIG_A2019123778
AFTER DELETE ON CONCELHO
FOR EACH ROW
DECLARE
LOCALREMOVIDO local.localvacinacao%TYPE;

Diogo Dias Lopes 2018019746
Leonardo David de Bastos Marques 2019123778
Ricardo Jorge Loureiro Rodrigues 2017015954
```

```
SELECT local.localvacinacao INTO LOCALREMOVIDO  
FROM CONCELHO,LOCAL  
WHERE concelho.id_concelho = local.id_concelho AND  
concelho.id_concelho=:OLD.ID_CONCELHO;
```

```
DELETE FROM local  
WHERE local.localvacinacao = LOCALREMOVIDO;
```

```
END;
```

Diogo Dias Lopes 2018019746

Leonardo David de Bastos Marques 2019123778

Ricardo Jorge Loureiro Rodrigues 2017015954

## parâmetros físicos

### tabela frasco

<b>INSERÇÕES</b>			
Cerca de 1/5 das seringas pois em media cada frasco da para 5 seringas			
<b>ACTUALIZAÇÕES</b>			
Algumas atualizações			
PF	PCTFREE	12%	[ 1..25 ]
PU	PCTUSED	88%	[ 40..90 ]
PINC	PCTINCREASE	10%	[ 0..100] DEFEITO 50
<b>ATRIBUTOS</b>			
	<b>NOME</b>	<b>TIPO</b>	<b>TAMANHO</b>
	idfrasco	NUMBER	4 BYTES
	idtransporte	NUMBER	4 BYTES
	idvacina	number	4 BYTES
	idlote	number	4 BYTES
	ndoses	number	4 BYTES
	data_aberto	date	7 BYTES
	local	char	20
	estado	char	10
STA	SOMA TOTAL ATRIBUTOS	57	BYTES
	Header Variavel (por registo)	5	BYTES
	Header de coluna (por registo)	8	BYTES
TMR	TAMANHO MEDIO DO REGISTO	70	BYTES
TB	TAMANHO DO BLOCO	4 098	BYTES
	POR DEFEITO : 4098 BYTES		
HF	HEADER FIXO (Máximo 107 Bytes)	84	BYTES
	+/- 8% do Bloco		
ELB	ESPAÇO LIVRE NO BLOCO	3 522	BYTES
	TB * (100% - PCTFREE) - HF		
NRB	NÚMERO DE REGISTO POR BLOCO	50	Registos
	ELB / TMR (ARREDONDADO PARA BAIXO)		
NRI	NÚMERO INICIAL DE REGISTOS	400000	Registos
NB	NÚMERO DE BLOCOS	8000	Blocos
EIT	ESPAÇO INICIAL DA TABELA	32784000	BYTES

Diogo Dias Lopes 2018019746

Leonardo David de Bastos Marques 2019123778

Ricardo Jorge Loureiro Rodrigues 2017015954

INSERÇÕES

Muitas inserções (quase as mesmas que os utentes)

ACTUALIZAÇÕES

Poucas atualizações ou escassas

PF

PCTFREE

5%

%

[ 1..25 ]

PU

PCTUSED

95

%

[ 40..90 ]

PINC

PCTINCREASE

10%

%

[ 0 .. 100 ]

DEFEITO 50

ATRIBUTOS

NOME	TIPO	TAMANHO	
Idconvocatoria	NUMBER	4	BYTES
Nfase	number	4	BYTES
id_toma	NUMBER	4	BYTES
data_convocatoria	date	7	BYTES
resposta	char	10	BYTES
localvacinacao	char	21	
STA	SOMA TOTAL ATRIBUTOS		50 BYTES
	Header Variavel (por registo)	5	BYTES
	Header de coluna (por registo)	6	BYTES
TMR	TAMANHO MEDIO DO REGISTO		61 BYTES

TB

TAMANHO DO BLOCO

4 096

BYTES

POR DEFEITO : 4098 BYTES

HF

HEADER FIXO (Máximo 107 Bytes)

84

BYTES

+/- 8% do Bloco

ELB

ESPAÇO LIVRE NO BLOCO

3 807

BYTES

TB \* (100% - PCTFREE) - HF

NRB

NÚMERO DE REGISTO POR BLOCO

62

Registos

ELB / TMR (ARREDONDADO PARA BAIXO)

NRI

NÚMERO INICIAL DE REGISTOS

1000000

Registos

NB

NÚMERO DE BLOCOS

16129

Blocos

EIT

ESPAÇO INICIAL DA TABELA

66064384

BYTES

Diogo Dias Lopes 2018019746

Leonardo David de Bastos Marques 2019123778

Ricardo Jorge Loureiro Rodrigues 2017015954

## Tabela utente

INSERÇÕES			
Muitas inserções cerca de 10M(se estivermos só a falar em Portugal)			
ACTUALIZAÇÕES			
Algumas			
PF	PCTFREE	12%	% [1..25]
PU	PCTUSED	88%	% [40..90]
PINC	PCTINCREASE	10%	% [0..100] DEFEITO 50
ATRIBUTOS			
NOME	TIPO	TAMANHO	
Id_utente	NUMBER	4	BYTES
Id_concelho	NUMBER	4	BYTES
Id_profissao	NUMBER	4	BYTES
Nome	Char	21	BYTES
Idade	NUMBER	4	BYTES
Imunizado	bool	1	BYTES
LocalDeTrabalho	char	21	BYTES
DataDenascimento	DATE	7	BYTES
rejeitou	bool	1	bytes
convocado	bool	1	bytes
ndosestomadas	number	4	bytes
dataultimadose	date	7	bytes
centro_saude	char	21	bytes
STA	SOMA TOTAL ATRIBUTOS	100	BYTES
	Header Variavel (por registo)	5	BYTES
	Header de coluna (por registo)	13	BYTES
TMR	TAMANHO MEDIO DO REGISTO	118	BYTES
TB	TAMANHO DO BLOCO	2048	BYTES
	POR DEFEITO :4098 BYTES		
HF	HEADER FIXO (Máximo 107 Bytes)	84	BYTES
	+/- 8% do Bloco		
ELB	ESPAÇO LIVRE NO BLOCO	1718	BYTES
	TB * (100% - PCTFREE) - HF		
NRB	NÚMERO DE REGISTO POR BLOCO	14	Registos
	ELB / TMR (ARREDONDADO PARA BAIXO)		
NRI	NÚMERO INICIAL DE REGISTOS	1000000	Registos
			começei inicialmente por considerar que 10% da população foi vacinada
NB	NÚMERO DE BLOCOS	71428	Blocos
EIT	ESPAÇO INICIAL DA TABELA	146284544	BYTES

Diogo Dias Lopes 2018019746

Leonardo David de Bastos Marques 2019123778

Ricardo Jorge Loureiro Rodrigues 2017015954

INSERÇÕES			
Muitas inserções o dobro dos utentes			
ACTUALIZAÇÕES			
Poucas atualizações ou escassas			
PF	PCTFREE	5%	[ 1 .. 25 ]
PU	PCTUSED	95%	[ 40 .. 90 ]
PINC	PCINCREASE	10%	[ 0 .. 100 ] DEFEITO 50
ATRIBUTOS			
NOME	TIPO	TAMANHO	
Idtoma	number	4	BYTES
Id_convocatoria	number	4	BYTES
Idutente	NUMBER	4	BYTES
Idturno	NUMBER	4	BYTES
datatoma	date	7	BYTES
compareceu	bool	1	BYTES
STA	SOMA TOTAL ATRIBUTOS	24	BYTES
	Header Variavel (por registo)	5	BYTES
	Header de coluna (por registo)	6	BYTES
TMR	TAMANHO MEDIO DO REGISTO	35	BYTES
TB	TAMANHO DO BLOCO	4 096	BYTES
POR DEFEITO : 4096 BYTES			
HF	HEADER FIXO (Máximo 107 Bytes)	84	BYTES
+/- 8% do Bloco			
ELB	ESPAÇO LIVRE NO BLOCO	3 807	BYTES
TB * (100% - PCTFREE) - HF			
NRB	NÚMERO DE REGISTO POR BLOCO	108	Registos
ELB / TMR (ARREDONDADO PARA BAIXO)			
NRI	NÚMERO INICIAL DE REGISTOS	2000000	Registos
NB	NÚMERO DE BLOCOS	18518	Blocos
EIT	ESPAÇO INICIAL DA TABELA	75843728	BYTES

Diogo Dias Lopes 2018019746

Leonardo David de Bastos Marques 2019123778

Ricardo Jorge Loureiro Rodrigues 2017015954

